Listing of Claims

1. (Original) A method of instantiating a device driver, comprising:

dynamically associating a first software component with the device driver at run-time, the first software component containing information that facilitates communication with devices of a specific device type.

2. (Original) A method as recited in Claim 1, further comprising:

defining a plurality of device parameters;

associating at least one of the plurality of device parameters with a service; and

communicating the at least one of the plurality of device parameters associated with the service to the device driver.

3. (Original) A method as recited in Claim 2, wherein defining the plurality of device parameters comprises:

declaring a parameter base class that defines the plurality of device parameters;

wherein associating the at least one of the plurality of device parameters with the service comprises:

deriving a service-specific sub-class from the base class that defines the at least one of the plurality of device parameters that are associated with the service;

wherein the method further comprises:

instantiating the service-specific sub-class to create a service-specific sub-class object; and

instantiating the parameter base class to create a parameter base class object.

4. (Original) A method as recited in Claim 3, wherein communicating the at least one of the plurality of device parameters associated with the service to the device driver comprises:

passing the at least one of the plurality of device parameters associated with the service from the service-specific sub-class object to the device driver.

5. (Original) A method as recited in Claim 1, further comprising:

defining a plurality of common device parameters;

defining a plurality of service-specific device parameters;

associating the common device parameters with the service-specific device parameters; and

communicating the common device parameters and the service-specific device parameters to the device driver.

6. (Original) A method as recited in Claim 5, wherein defining the plurality of common device parameters comprises:

declaring a parameter base class that defines the plurality of common device parameters;

wherein defining the plurality of service-specific device parameters comprises:

providing a second software component that comprises one of a script file and an extensible markup language (XML) file; and

wherein the method further comprises:

instantiating the parameter base class to create a parameter base class object.

7. (Original) A method as recited in Claim 6, wherein associating the common device parameters with the service-specific device parameters comprises:

dynamically loading the parameter base class object with the second software component at run time.

8. (Original) A method as recited in Claim 7, wherein communicating the common device parameters and the service-specific device parameters to the device driver comprises:

passing the common device parameters and the service-specific device parameters from the parameter base class object to the device driver after loading the parameter base class object with the second software component at run time.

9. (Original) A method as recited in Claim 1, wherein the first software component comprises one of a script file and an extensible markup language (XML) file.

10. (Original) A method as recited in Claim 1, wherein dynamically associating the first software component with the device driver at run-time comprises:

selecting the first software component from a plurality of software components, respective ones of the plurality of software components being associated with respective ones of a plurality of device types.

11. (Original) A method as recited in Claim 10, further comprising:

generating the plurality of software components based on a plurality of management information base (MIB) files, respective ones of the plurality of MIB files being associated with respective ones of the plurality of device types.

12. (Original) A method of collecting data from a device, comprising:

receiving a request to collect data from the device;

dynamically associating a software component with a device driver at run-time, the software component containing information that facilitates communication with the device; and

retrieving data from the device using the device driver.

13. (Original) A method as recited in Claim 12, wherein retrieving data from the device using the device driver comprises:

associating at least one device parameter with a service;

communicating the at least one device parameter to the device driver; and

retrieving data associated with the at least one device parameter from the device.

14. (Original) A method as recited in Claim 12, wherein the first software component comprises one of a script file and an extensible markup language (XML) file.

15. (Original) A method as recited in Claim 12, wherein dynamically associating the software component with the device driver at run-time comprises:

selecting the first software component from a plurality of software components, respective ones of the plurality of software components being associated with respective ones of a plurality of device types.

16. (Original) A method of instantiating a device driver, comprising:

defining a plurality of device parameters;

associating at least one of the plurality of device parameters with a service; and

dynamically communicating the at least one of the plurality of device parameters associated with the service to the device driver at run-time.

17. (Original) A method as recited in Claim 16, wherein defining the plurality of device parameters comprises:

defining a plurality of common device parameters; and

defining a plurality of service-specific device parameters.

18. (Original) A method as recited in Claim 17, wherein associating the at least one of the plurality of device parameters with the service comprises:

associating the common device parameters with the service-specific device parameters; and

wherein dynamically communicating the at least one of the plurality of device parameters associated with the service to the device driver at run-time comprises:

dynamically communicating the common device parameters and the service-specific device parameters to the device driver at run-time.

19. (Original) A method as recited in Claim 16, wherein defining the plurality of device parameters comprises:

declaring a parameter base class that defines the plurality of device parameters; and

wherein associating the at least one of the plurality of device parameters with the service comprises:

deriving a service-specific sub-class from the base class that defines the at least one of the plurality of device parameters that are associated with the service.

20. (Original) A system for instantiating a device driver, comprising:

means for dynamically associating a first software component with the device driver at run-time, the first software component containing information that facilitates

communication with devices of a specific device type.

21. (Original) A system as recited in Claim 20, further comprising:

means for defining a plurality of device parameters;

means for associating at least one of the plurality of device parameters with a service; and

means for communicating the at least one of the plurality of device parameters associated with the service to the device driver.

22. (Original) A system as recited in Claim 21, wherein the means for defining the plurality of device parameters comprises:

means for declaring a parameter base class that defines the plurality of device parameters;

wherein the means for associating the at least one of the plurality of device parameters with the service comprises:

means for deriving a service-specific sub-class from the base class that defines the at least one of the plurality of device parameters that are associated with the service;

wherein the system further comprises:

means for instantiating the service-specific sub-class to create a service-specific sub-class object; and

means for instantiating the parameter base class to create a parameter base class object.

23. (Original) A system as recited in Claim 22, wherein the means for communicating the at least one of the plurality of device parameters associated with the service to the device driver comprises:

means for passing the at least one of the plurality of device parameters associated with the service from the service-specific sub-class object to the device driver.

24. (Original) A system as recited in Claim 20, further comprising:

means for defining a plurality of common device parameters;

means for defining a plurality of service-specific device parameters;

means for associating the common device parameters with the service-specific device parameters; and

means for communicating the common device parameters and the service-specific device parameters to the device driver.

25. (Original) A system as recited in Claim 24, wherein the means for defining the plurality of common device parameters comprises:

means for declaring a parameter base class that defines the plurality of common device parameters;

wherein the means for defining the plurality of service-specific device parameters comprises:

means for providing a second software component that comprises one of a script file and an extensible markup language (XML) file; and

wherein the system further comprises:

means for instantiating the parameter base class to create a parameter base class object.

26. (Original) A system as recited in Claim 25, wherein the means for associating the common device parameters with the service-specific device parameters comprises:

means for dynamically loading the parameter base class object with the second software component at run time.

27. (Original) A system as recited in Claim 26, wherein the means for communicating the common device parameters and the service-specific device parameters to the device driver comprises:

means for passing the common device parameters and the service-specific device parameters from the parameter base class object to the device driver after loading the parameter base class object with the second software component at run time.

28. (Original) A system as recited in Claim 20, wherein the first software component comprises one of a script file and an extensible markup language (XML) file.

29. (Original) A system as recited in Claim 20, wherein the means for dynamically associating the first software component with the device driver at run-time comprises:

means for selecting the first software component from a plurality of software components, respective ones of the plurality of software components being associated with respective ones of a plurality of device types.

30. (Original) A system as recited in Claim 29, further comprising:

means for generating the plurality of software components based on a plurality of management information base (MIB) files, respective ones of the plurality of MIB files being associated with respective ones of the plurality of device types.

31. (Original) A system for collecting data from a device, comprising:

means for receiving a request to collect data from the device;

means for dynamically associating a software component with a device driver at run-time, the software component containing information that facilitates communication with the device; and

means for retrieving data from the device using the device driver.

32. (Original) A system as recited in Claim 31, wherein the means for retrieving data from the device using the device driver comprises:

means for associating at least one device parameter with a service;

means for communicating the at least one device parameter to the device driver; and

means for retrieving data associated with the at least one device parameter from the device.

33. (Original) A system as recited in Claim 31, wherein the first software component comprises one of a script file and an extensible markup language (XML) file.

34. (Original) A system as recited in Claim 31, wherein the means for dynamically associating the software component with the device driver at run-time comprises:

means for selecting the first software component from a plurality of software components, respective ones of the plurality of software components being associated with

respective ones of a plurality of device types.

35. (Original) A system for instantiating a device driver, comprising:

means for defining a plurality of device parameters;

means for associating at least one of the plurality of device parameters with a service; and

means for dynamically communicating the at least one of the plurality of device parameters associated with the service to the device driver at run-time.

36. (Original) A system as recited in Claim 35, wherein the means for defining the plurality of device parameters comprises:

means for defining a plurality of common device parameters; and

means for defining a plurality of service-specific device parameters.

37. (Original) A system as recited in Claim 36, wherein the means for associating the at least one of the plurality of device parameters with the service comprises:

means for associating the common device parameters with the service-specific device parameters; and

wherein the means for dynamically communicating the at least one of the plurality of device parameters associated with the service to the device driver at run-time comprises:

means for dynamically communicating the common device parameters and the service-specific device parameters to the device driver at run-time.

38. (Original) A system as recited in Claim 35, wherein the means for defining the plurality of device parameters comprises:

means for declaring a parameter base class that defines the plurality of device parameters; and

wherein the means for associating the at least one of the plurality of device parameters with the service comprises:

means for deriving a service-specific sub-class from the base class that defines the at least one of the plurality of device parameters that are associated with the service.

39. (Original) A computer program product for instantiating a device driver, comprising:

a computer readable storage medium having computer readable program code embodied therein, the computer readable program code comprising:

computer readable program code for dynamically associating a first software component with the device driver at run-time, the first software component containing information that facilitates communication with devices of a specific device type.

40. (Original) A computer program product as recited in Claim 39, further comprising:

computer readable program code for defining a plurality of device parameters;

computer readable program code for associating at least one of the plurality of device parameters with a service; and

computer readable program code for communicating the at least one of the plurality of device parameters associated with the service to the device driver.

41. (Original) A computer program product as recited in Claim 40, wherein the computer readable program code for defining the plurality of device parameters comprises:

computer readable program code for declaring a parameter base class that defines the plurality of device parameters;

wherein the computer readable program code for associating the at least one of the plurality of device parameters with the service comprises:

computer readable program code for deriving a service-specific sub-class from the base class that defines the at least one of the plurality of device parameters that are associated with the service;

wherein the computer program product further comprises:

computer readable program code for instantiating the service-specific sub-class to create a service-specific sub-class object; and

computer readable program code for instantiating the parameter base class to create a parameter base class object.

42. (Original) A computer program product as recited in Claim 41, wherein the

computer readable program code for communicating the at least one of the plurality of device parameters associated with the service to the device driver comprises:

computer readable program code for passing the at least one of the plurality of device parameters associated with the service from the service-specific sub-class object to the device driver.

43. (Original) A computer program product as recited in Claim 39, further comprising:

computer readable program code for defining a plurality of common device parameters;

computer readable program code for defining a plurality of service-specific device parameters;

computer readable program code for associating the common device parameters with the service-specific device parameters; and

computer readable program code for communicating the common device parameters and the service-specific device parameters to the device driver.

44. (Original) A computer program product as recited in Claim 43, wherein the computer readable program code for defining the plurality of common device parameters comprises:

computer readable program code for declaring a parameter base class that defines the plurality of common device parameters;

wherein the computer readable program code for defining the plurality of service-specific device parameters comprises:

computer readable program code for providing a second software component that comprises one of a script file and an extensible markup language (XML) file; and

wherein the computer program product further comprises:

computer readable program code for instantiating the parameter base class to create a parameter base class object.

45. (Original) A computer program product as recited in Claim 44, wherein the computer readable program code for associating the common device parameters with the

service-specific device parameters comprises:

computer readable program code for dynamically loading the parameter base class object with the second software component at run time.

46. (Original) A computer program product as recited in Claim 45, wherein the computer readable program code for communicating the common device parameters and the service-specific device parameters to the device driver comprises:

computer readable program code for passing the common device parameters and the service-specific device parameters from the parameter base class object to the device driver after loading the parameter base class object with the second software component at run time.

47. (Original) A computer program product as recited in Claim 39, wherein the first software component comprises one of a script file and an extensible markup language (XML) file.

48. (Original) A computer program product as recited in Claim 39, wherein the computer readable program code for dynamically associating the first software component with the device driver at run-time comprises:

computer readable program code for selecting the first software component from a plurality of software components, respective ones of the plurality of software components being associated with respective ones of a plurality of device types.

49. (Original) A computer program product as recited in Claim 48, further comprising:

computer readable program code for generating the plurality of software components based on a plurality of management information base (MIB) files, respective ones of the plurality of MIB files being associated with respective ones of the plurality of device types.

50. (Original) A computer program product for collecting data from a device, comprising:

a computer readable storage medium having computer readable program code embodied therein, the computer readable program code comprising:

computer readable program code for receiving a request to collect data from the device;

computer readable program code for dynamically associating a software component with a device driver at run-time, the software component containing information that facilitates communication with the device; and

computer readable program code for retrieving data from the device using the device driver.

51. (Original) A computer program product as recited in Claim 50, wherein the computer readable program code for retrieving data from the device using the device driver comprises:

computer readable program code for associating at least one device parameter with a service;

computer readable program code for communicating the at least one device parameter to the device driver; and

computer readable program code for retrieving data associated with the at least one device parameter from the device.

52. (Original) A computer program product as recited in Claim 50, wherein the first software component comprises one of a script file and an extensible markup language (XML) file.

53. (Original) A computer program product as recited in Claim 50, wherein the computer readable program code for dynamically associating the software component with the device driver at run-time comprises:

computer readable program code for selecting the first software component from a plurality of software components, respective ones of the plurality of software components being associated with respective ones of a plurality of device types.

54. (Original) A computer program product for instantiating a device driver, comprising:

a computer readable storage medium having computer readable program code

embodied therein, the computer readable program code comprising:

computer readable program code for defining a plurality of device parameters;

computer readable program code for associating at least one of the plurality of device parameters with a service; and

computer readable program code for dynamically communicating the at least one of the plurality of device parameters associated with the service to the device driver at run-time.

55. (Original) A computer program product as recited in Claim 54, wherein the computer readable program code for defining the plurality of device parameters comprises:

computer readable program code for defining a plurality of common device parameters; and

computer readable program code for defining a plurality of service-specific device parameters.

56. (Original) A computer program product as recited in Claim 55, wherein the computer readable program code for associating the at least one of the plurality of device parameters with the service comprises:

computer readable program code for associating the common device parameters with the service-specific device parameters; and

wherein the computer readable program code for dynamically communicating the at least one of the plurality of device parameters associated with the service to the device driver at run-time comprises:

computer readable program code for dynamically communicating the common device parameters and the service-specific device parameters to the device driver at run-time.

57. (Original) A computer program product as recited in Claim 54, wherein the computer readable program code for defining the plurality of device parameters comprises:

computer readable program code for declaring a parameter base class that defines the plurality of device parameters; and

wherein the computer readable program code for associating the at least one of the plurality of device parameters with the service comprises:

computer readable program code for deriving a service-specific sub-class from the

base class that defines the at least one of the plurality of device parameters that are associated with the service.